URC FUZZY MODELING AND SIMULATION OF GENE REGULATION

B. A. Sokhansanj[1,2] and J. P. Fitch[1]
[1]Biology and Biotechnology Research Program, Lawrence Livermore National Laboratory, Livermore, CA, USA
[2]Department of Applied Science (Livermore), University of California, Davis, CA, USA

Abstract— **Recent technological advances in high-throughput data collection give biologists the ability to study increasingly complex systems. A new methodology is needed to develop and test biological models based on experimental observations and predict the effect of perturbations of the network (e.g. genetic engineering, pharmaceuticals, gene therapy). Diverse modeling approaches have been proposed, in two general categories: modeling a biological pathway as (a) a logical circuit or (b) a chemical reaction network. Boolean logic models can not represent necessary biological details. Chemical kinetics simulations require large numbers of parameters that are very difficult to accurately measure. Based on the way biologists have traditionally thought about systems, we propose that fuzzy logic is a natural language for modeling biology. The Union Rule Configuration (URC) avoids combinatorial explosion in the fuzzy rule base, allowing complex system models. We demonstrate the fuzzy modeling method on the commonly studied *lac* operon of *E. coli*. Our goal is to develop a modeling and simulation approach that can be understood and applied by biologists without the need for experts in other fields or "black-box" software.**

Keywords— **gene function, gene regulation, fuzzy logic, simulation, modeling, microbiology**

I. Introduction

Technological advances in DNA sequencing [1] have made it feasible to obtain the entire genetic sequence (genome) of an organism being studied by biologists. While the genomes of plants and animals are generally large (10^8-10^{10} bases, $O(10^4)$ genes) and still take months and years to sequence, it is now possible to generate the draft genome sequence of a bacterium (10^6 bases, $O(10^3)$ genes) in a matter of days or even hours. However, the sequence of genes only provides a "parts list" for the cell. Cell function arises from the regulatory pathways and networks of the genes and their protein products: how the parts are assembled and work together in response to environmental stimuli. This regulation is very complex, and involves protein-protein and protein-DNA interactions in response to environmental effects, with multiple feedback as illustrated in Fig. 1.

We are now in the "Post-Sequencing" era of biotechnology, characterized by engineering advances (reviewed in [1]) such as DNA chips and microarrays for mRNA transcript profiling, high-throughput X-ray and NMR spectroscopy coupled with computational techniques for protein structure determination, and protein profiling with mass spectroscopy and 2-D gel electrophoresis. The purpose of these technologies is to study an entire network

This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory under Contract No.W-7405-Eng-48.

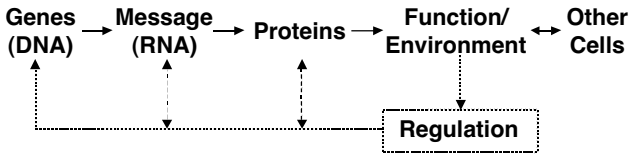

Fig. 1. Coding segments of DNA sequences (genes) are transcribed on to message RNA (mRNA). The mRNA is then translated to the proteins that perform cellular functions. Regulatory feedback occurs at any of step.

(or "pathway") of genes, proteins, and biochemical reactions dedicated to performing a particular function of the cell, usually in response to some environmental stimulus. Examples for bacteria include nutrient metabolism, infection of a host, spore formation, DNA repair, etc. By extension, the ultimate goal is to look at all the interconnected networks of the living cell as a whole. Given the complexity of systems being studied, biologists need a modeling and simulation framework to make sense of large-scale data and intelligently design traditional bench-top experiments that provide the most biological insight.

II. Modeling & Simulation in Biology

Typically, biologists qualitatively model the systems they observe, which they describe in text or a diagram (e.g. description of lac operon below and Fig. 2). The model is developed and confirmed by experiments that test a specific hypothesis. While experimental results may be analyzed quantitatively (e.g. enzyme kinetic assay) or qualitatively (e.g. protein gel electrophoresis), the conclusion is always qualitative: a component or connection is added or removed from the model, and given a linguistic description like "weak inhibition" or "strongly expressed". Quantitative conclusions are avoided, because while results may be sufficiently internally consistent to support, for example, curve fitting, they can not be used with results from separate experiments because of large sources of error. For the same enzymatic reaction, different experiments may provide kinetic parameters that vary over 1-2 orders of magnitude. Reliable measurement requires careful, time-consuming and costly experiments that do not provide *any new scientific insight*. However, these measurements are performed on a limited set of parameters required for engineering applications like pharmacology, biological risks of toxin exposure, and biocatalysts for enzyme production.

As systems become more complex, qualitative "mental" models become harder to develop and interpret. There have been numerous computer modeling and simulation methods proposed over the past 40 years. There are two general types. One type is a (typically) Boolean network model (e.g.

Report Documentation Page

Report Date	Report Type	Dates Covered (from... to)
25 Oct 2001	N/A	-

Title and Subtitle URC Fuzzy Modeling and Simulation of Gene Regulation	**Contract Number**
	Grant Number
	Program Element Number
Author(s)	**Project Number**
	Task Number
	Work Unit Number
Performing Organization Name(s) and Address(es) Biology and Biotechnology Research Program Lawrence Livermore National Laboratory Livermore, CA	**Performing Organization Report Number**
Sponsoring/Monitoring Agency Name(s) and Address(es) sponsoring agency and address	**Sponsor/Monitor's Acronym(s)**
	Sponsor/Monitor's Report Number(s)

Distribution/Availability Statement
Approved for public release, distribution unlimited

Supplementary Notes
Papers from 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, October 25-28, 2001, held in Istanbul, Turkey. See also ADM001351 for entire conference on cd-rom., The original document contains color images. , The original document contains color images.

Abstract

Subject Terms

Report Classification unclassified	**Classification of this page** unclassified
Classification of Abstract unclassified	**Limitation of Abstract** UU

Number of Pages
3

[2]). For example, if a gene is active, it is given a value of 1. Boolean relations define state transitions based on different combinations of active genes. This model is easy to implement, and it has yielded many useful insights into the self-organization of chaotic systems. However, it does not have adequate resolution to model biological systems, since they often depend on continuously variable quantities and interaction strengths. The other type of model is defining the biological system as a network of coupled chemical reactions, and then simulating those reactions (e.g. [3]). This requires both experimental parameters and an assumption of a reaction mechanism. Most often, the simulation consists of solving differential equations based on Michaelis-Menten kinetics; recently stochastic reaction simulations have been used as well [4]. However, as described above, it is very difficult to obtain reliable kinetic data. At least some parameters are simply guessed. To obtain these parameters would require many very difficult and costly experiments with limited immediate scientific value, in a field where so many basic questions are still unanswered.

We propose that *fuzzy logic is a natural language for modeling biology*. Fuzzy set theory is based on normal set theory, but fuzzy set membership may range anywhere from 0 (absolutely outside set) through 1 (absolutely inside set). Fuzzy logic was introduced by Zadeh in 1965 [5] and is now being actively used to model control systems and business processes; there are countless papers [6] and textbooks (e.g. [7]) reviewing theory and applications. Early in its development, fuzzy logic was suggested as the basis for linguistic modeling [8]: fuzzy sets representing words like "low" or "strong" are used instead of numbers to describe quantities being modeled. Thus, fuzzy logic can be used to formalize how biology is currently modeled, as well as to provide a basis for computer simulation. Fuzzy logic has been applied to modeling complex biochemical reactions [9], modeling biocatalyst control systems [10], and microarray data analysis [11]. We propose expanding these efforts to a broader application of fuzzy logic to gene regulation.

III. Fuzzy Modeling Methodology

A general discussion of fuzzy modeling is outside the scope of this paper (see [5]-[8] and others), so we will focus on methodology for the gene regulation problem specifically. In our work, we represent a fuzzy quantity with membership from 0 to 1.0 in five sets, {VL, LO, ME, HI, VH}. For example, a "low, non-negligible" quantity could be

$$\mathbf{P} = \{ (VL, 0.2), (LO, 0.9), (ME, 0.1), (HI, 0), (VH, 0) \}$$

If quantitative data is known, it is "fuzzified" into this form. Fig. 3 shows how a graph of fuzzy set domains defined for lac enzyme concentration. For another protein with a different natural concentration range, e.g. the lac repressor, a different fuzzification scheme is defined. A fuzzy quantity is "defuzzified" by taking the centroid of the area of the fuzzification graph defined by the membership functions.

In our models, both quantity *and* activity level are important. There is a fixed number of promoters of a gene (unless it has been genetically engineered), but their strength may vary. Proteins have two associated variables: production level and activity. Production level is transient, but even after proteins are produced they remain functional until deactivated or decayed. Protein activity is affected by production level, but also by decay by proteases, interactions with other proteins and chemicals, and environmental conditions For example, lac repressor protein activity is inhibited by lactose. Like with control systems, activity is either fuzzified based on assumptions from observational data about the effect of perturbing activity level (e.g. by introducing a genetic mutation).

The fuzzy forms of operators like AND and OR can be implemented in different ways. In the simplest form, **P** AND **Q** takes the minimum of the membership of **P** and **Q**

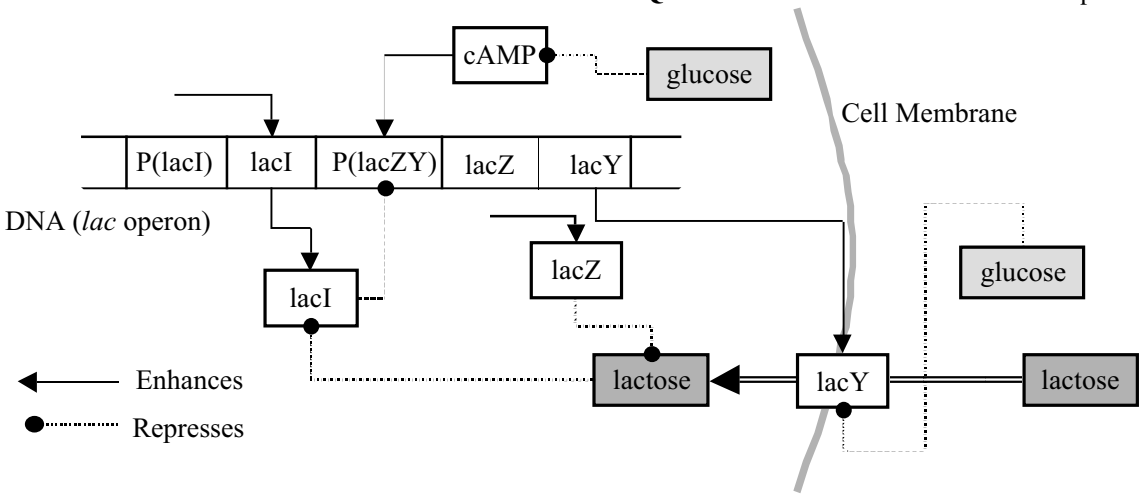

Fig. 2. Model of *lac* operon regulation, showing inhibition and activation of enzymes, substrates, and regulators. Proteins produced by lacZ and lacY are labeled with gene names. The lacZ enzyme is shown to be "repressing" lactose, in fact it is breaking it down into glucose and galactose. See text for description.

in each fuzzy set. OR takes the maximum. In another implementation suitable for modeling, AND is a product of memberships and OR is a sum. In both of these (but not all) implementations, IF **P** THEN **Q** is synonymous with AND.

A typical node of a fuzzy model has N inputs and one output, i.e.

$$\text{IF } \mathbf{P}_1 \text{ AND } \mathbf{P}_2 \text{ AND } \mathbf{P}_3 \text{ AND } \ldots \text{ AND } \mathbf{P}_N \text{ THEN } \mathbf{Q}$$

If the inputs $\mathbf{P}_i$ have M fuzzy sets, this requires a rule base with M^N rules, e.g. a rule for (VL) AND (VL) AND (VL)..., another rule for (LO) AND (VL) AND (VL)..., etc. If we want to model the production level of a protein that depends on the strength of two promoters, translation rate, temperature, and 5 regulatory proteins, this requires 5^9 = 1,953,125 rules! The most common solution is clustering variables. While clustering is certainly useful in combining proteins that are co-regulated, as identified from microarray experiments for example, it defeats the purpose of a reasonably detailed gene regulation model.

Recently, Combs ([7]-Appendix, [12]) proposed a solution to the "curse of dimensionality" called the Union Rule Configuration (URC). In this configuration, the above node would instead be written as

$$(\text{IF } \mathbf{P}_1 \text{ THEN } \mathbf{Q}) \text{ OR } (\text{IF } \mathbf{P}_2 \text{ THEN } \mathbf{Q}) \text{ OR } \ldots (\text{IF } \mathbf{P}_N \text{ THEN } \mathbf{Q})$$

This form now only requires M*N rules, or in our example 5*9 = 45 rules. Not only is rule evaluation computationally feasible, but mining data to obtain the rule base can be done very quickly using conventional algorithms. Fig. 4 shows the URC rule base for the lac operon model. In addition, rules can also have weights that multiply the consequent membership. Weighting rules lets our model distinguish between strong and weak interactions.

Despite its advantages, the URC remains controversial, since it is likely not equivalent to the original formulation in fuzzy logic (they are in classical logic), so past rigorous proofs do not necessarily apply. However, the URC succeeds as a heuristic method in a number of different problems, and we have used it for our biological models.

IV. Implementation— Lac Operon

The most studied gene regulation system is the lac operon (illustrated schematically in Fig. 2) of the bacterium *E. coli*. The operon is a prototype for most genetic regulatory systems in bacteria, in that it involves a group of genes regulated together by one or two stimuli. Regulation in plant and animal cells is generally more complex. We will provide a brief overview here, but the lac operon is described in detail in any introductory biochemistry or molecular biology textbook (e.g. [13]).

Generally, *E. coli* and similar bacteria use glucose from the surrounding environment as their source of energy. However, when lactose is available, the cell will draw it through its membrane and then break it down to glucose. This process is particularly favored when there is a shortage of glucose. The lac operon is a genetic program that implements this regulation. It consists of four genes and a number of protein binding sites clustered near each other in the *E. coli* chromosome. The genes and their protein products are lacI (lac repressor), lacZ (ß-galactosidase or lac enzyme), lacY (ß-galactoside permease or lac permease), and lacA (not involved in lactose regulation). Lac permease transports lactose into the cell and the lac enzyme breaks it down to glucose and galactose. When RNA polymerase (RNAP) binds to the *promoter* of the gene (labeled P(lacI) and P(lacZY) in Fig. 2), it catalyzes its transcription. Promoters have different binding strengths; for example, lacI has a very weak promoter (due to a non-optimal DNA sequence for protein binding). The rule IF (promoter strength) THEN (protein production) refers to the absolute promoter strength independent of any other regulatory activity (which is modeled with separate IF/THEN rules). This allows us to model the effect of modifying the promoter on protein production, a common genetic engineering technique.

Dynamic regulation results from protein binding to *operator* sites near the promoter. When RNAP binds to the promoter, it has to travel down the DNA to transcribe the lacZ and lacY genes. Lac repressor can bind to operators just past the promoter, and block the RNA polymerase. When there is lactose present in the cell, it prevents lac repressor from blocking RNAP. We model this as inhibition of lac repressor activity. After lac enzyme consumes all the lactose, the repressor again binds to the operator, production from lacZ and lacY ceases, and eventually proteases degrade remaining lac enzymes. Thus, the lac operon is controlled by negative feedback. Additional regulation is provided by glucose. Glucose inhibits lac permease (lacY) activity to prevent lactose entry to the cell. It also inhibits cAMP, which binds (in conjunction with CRP) to an operator near P(lacZY). When cAMP is bound, it enhances production from the lacZ and lacY genes.

The proteins and sugars are all fuzzified on different domains, since they are present in different quantities. For example, Fig. 3 shows the domain of lac enzyme. This fuzzification is based on the known result that upon strong induction, 6.6% of bacterial protein mass is the enzyme. We

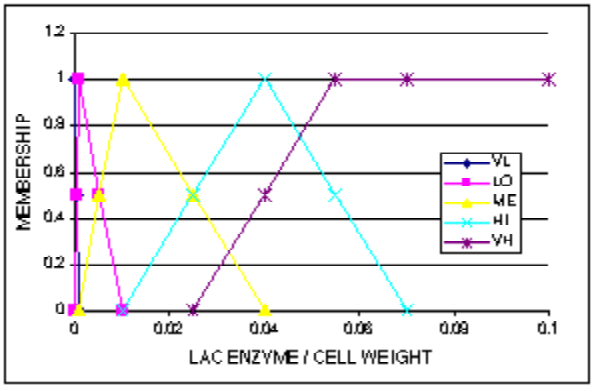

Fig. 3. Fuzzification of lac enzyme concentration, relative to total cell protein mass. The definition of VL is hidden to the left of LO.

TABLE 1
URC FUZZY RULE BASE FOR LAC OPERON

IF	VL	LO	ME	HI	VH
	lacI production				
P(lacI) strength	VL	LO	ME	HI	VH
	lacI activity				
lacI production	VL	LO	ME	HI	VH
lactose (in cell)	VH	ME	LO	LO	VL
protease	VH	HI	HI	LO	VL
	cAMP activity				
glucose	VH	HI	ME	LO	VL
	lacY, lacZ production				
P(lacZY) strength	VL	LO	ME	HI	VH
lacI activity	VH	HI	LO	LO	VL
cAMP activity	VL	LO	HI	HI	VH
	lacY activity				
lacY production	VL	LO	ME	HI	VH
glucose	VH	HI	ME	LO	LO
protease	VH	HI	HI	LO	VL
	lacZ activity				
lacZ production	VL	LO	ME	HI	VH
protease	VH	HI	HI	LO	VL
	lactose (in cell)				
lactose (outside cell)	VL	LO	ME	HI	VH
lacY activity	VL	LO	ME	HI	VH
lacZ activity	VH	HI	ME	LO	VL

use units normalized to cell mass because it is constantly growing as it consumes nutrients. Lac permease production is at half the rate of lac enzyme, so it is modeled with the same fuzzy rules but defuzzified over a domain with concentrations half as large as for lac enzyme. When numerical parameters are available, variables may be defuzzified at any point in a simulation and normal kinetic equations can be solved. This can be useful for more realistically modeling protease activity, for example. Table 1 shows the basic URC fuzzy rule base (unweighted) for the lac operon (proteins are identified by their gene names, i.e. lacI = lac repressor).

We observe the expected pattern of rapid lac enzyme production upon addition of extracellular lactose, followed by decay over time as the extracellular lactose is exhausted. In the absence of any lac permease in the cell there is no processing of lactose, however even a very low quantity leads to some lactose entering and lac enzyme being produced. This is an effect that can not be modeled with Boolean variables, since they can only represent the absence or presence of a quantity. Using weights, we can also model the effect of adding substances that induce the lac operon even more strongly than lactose, like the lac inducer IPTG.

VI. CONCLUSIONS

The lac operon is one of the simplest systems in biology, though it continues to yield interesting experimental questions. Other bacterial systems can be modeled in the same way. For example, we are currently integrating these modeling techniques in our lab's study of the virulence pathway of *Yersinia pestis*, the bacterium that causes plague. In general, fuzzy models can be used for:

1) *Computer simulation of complex systems*. Beyond a certain system size, predicting the effect of a perturbation on a system or interpreting the observed outcome of that perturbation requires computer simulation. The simulation can also help make artificial modifications to gene circuits and enhance biological production, build vectors for gene therapy, etc. A major advantage is that the language of the simulation uses linguistic terms familiar to practicing biologists.

2) *Hybrid quantitative/qualitative models*. Given current and projected technologies, confident quantitative measurements will remain difficult. However, there are many cases when quantitative predictions are required or parameters are available, thus a modeling framework should be flexible.

3) *Regulatory inference from large-scale data*. A URC fuzzy model has a rule base that scales linearly with number of inputs. Thus, it becomes relatively easy to mine integrated sets of thousands of data points for gene regulation rules. Biologists can use this "rough draft" to develop hypotheses that can be tested experimentally to develop a more complete and confident model.

REFERENCES

[1] J. P. Fitch and Sokhansanj, B., "Genomic engineering: moving beyond DNA sequence to function," *Proc. IEEE*, vol. 88, pp. 1949-1971, Dec. 2000.

[2] S. Liang, Fuhrman, S., and Somogyi, R. "REVEAL, a general reverse engineering algorithm for inference of genetic network architectures," *Pacific Symposium on Biocomputing*, vol. 3, pp. 18-29, 2000. [Online]. http:// www-smi.stanford.edu/projects/helix/psb98/

[3] D. Endy, You, L, Yin, J., and Molineux, I. J., "Computation, prediction, and experimental tests of fitness for bacteriophage T7 mutants with permuted genomes," *Proc. Natl. Acad. Sci. USA*, vol. 97, pp. 5375-5380, May 9, 2000.

[4] A. Arkin, Ross, J., and McAdams, H. H., "Stochastic kinetic analysis of developmental pathway bifurcation in phage λ-infected *Escherichia coli* cells," *Genomics*, vol. 149, pp. 1633-1648, Aug. 1998.

[5] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338-352, 1965.

[6] J. M. Mendel, "Fuzzy logic systems for engineering: a tutorial," *Proc. IEEE*, vol. 83, pp. 345-377, March 1995.

[7] H. Zimmerman, *Fuzzy Set Theory — and its Applications*, 2nd ed., Boston: Kluwer, 1999.

[8] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. Syst. Man. Cybernet.*, vol. 3, pp. 28-44, Jan. 1973.

[9] Lee, B., Yen, J., Yang, L., and Liao, J. C., "Incorporating qualitative knowledge in enzyme kinetic models using fuzzy logic," *Biotechnol. Bioeng.*, vol. 62, Mar. 20, 1999.

[10] B. Ruggeri, Sassi, G., and Bosco, F., "Macro approach and fuzzy modeling of entrapped biocatalyst," *Biotechnol. Prog.*, vol. 16, pp. 44-51, Feb., 2000.

[11] Woolf, P. J., and Wang, Y., "A fuzzy logic approach to analyzing gene expression data," *Physiol. Genomics*, vol. 3, pp. 9-15, Jan.-Feb., 2000.

[12] W. E. Combs, and Andrews, J. E., "Combinatorial rule explosion eliminated by a fuzzy rule configuration," *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 1-11, Feb. 1998.

[13] M. Watson, J. D. Hopkins, N. H. Roberts, J. W. Steitz, J. A., and A. M. Weiner, *Molecular Biology of the Gene*, 4th ed., Menlo Park, CA: Benjamin/Cummings, 1987.